Parameterized Complexity of the Workflow Satisfiability Problem

Anders Yeo

Department of Mathematics
University of Johannesburg

Joint work with **Jason Crampton** and **Gregory Gutin**

July 25, 2012

Outline

Outline

1. Introduction and Known Results
2. New Results Overview
3. Outline of proofs
4. Hierarchies

A simple constrained workflow for purchase order processing

We are given a set of users, U, say u_1 =Sheldon Cooper, u_2 =Leonard Hofstadter, u_3 =Howard Wolowitz and u_4=Rajesh Koothrappali.

A simple constrained workflow for purchase order processing

We are given a set of users, U, say u_1 =Sheldon Cooper, u_2 =Leonard Hofstadter, u_3 =Howard Wolowitz and u_4=Rajesh Koothrappali.

We are given a set of steps, S, such as the following.

s_1 create purchase order	s_2 approve purchase order
s_3 sign goods received note	s_4 create payment
s_5 countersign goods received note	s_6 approve payment

Authorizations and Constraints

We are given a list, A, of who are authorized for which steps, such as $A(s_1) = \{u_1, u_2\}$, $A(s_2) = \{u_1, u_3, u_4\}$, ...

Authorizations and Constraints

We are given a list, A, of who are authorized for which steps, such as $A(s_1) = \{u_1, u_2\}$, $A(s_2) = \{u_1, u_3, u_4\}$, $\ldots$

Finally we are given a set of constraints, (ρ, S_1, S_2), where $S_1, S_2 \subseteq S$ and ρ is a binary relations.

Authorizations and Constraints

We are given a list, A, of who are authorized for which steps, such as $A(s_1) = \{u_1, u_2\}$, $A(s_2) = \{u_1, u_3, u_4\}$, ...

Finally we are given a set of constraints, (ρ, S_1, S_2), where $S_1, S_2 \subseteq S$ and ρ is a binary relations.

A constraint is satisfied if there exists $s_1 \in S_1$ and $s_2 \in S_2$, such that $(u(s_1), u(s_2)) \in \rho$, where $u(s)$ is the user assigned to step s.

For example: $(=, \{s_2\}, \{s_1, s_3\})$.

Full example

$U = \{u_1, u_2, u_3, u_4, u_5\}$ and $S = \{s_1, s_2, s_3\}$.

$A(s_1) = \{u_1, u_2\}$, $A(s_2) = \{u_1, u_3\}$ and $A(s_3) = \{u_1, u_4, u_5\}$

$C = \{(=, \{s_1\}, \{s_2, s_3\}), (\neq, \{s_2\}, \{s_3\})\}$.

Is there a solution?

Full example

$U = \{u_1, u_2, u_3, u_4, u_5\}$ and $S = \{s_1, s_2, s_3\}$.

$A(s_1) = \{u_1, u_2\}$, $A(s_2) = \{u_1, u_3\}$ and $A(s_3) = \{u_1, u_4, u_5\}$

$C = \{(=, \{s_1\}, \{s_2, s_3\}), (\neq, \{s_2\}, \{s_3\})\}$.

Is there a solution? YES.

One Solution: u_1 does s_1, u_1 does step s_2 and u_5 does step s_3.

Workflow Satisfiability Problem (WSP)

Crampton considered the case where in all constraints (ρ, S_1, S_2), S_1 and S_2 are singletons (and there were some extra restrictions).

Workflow Satisfiability Problem (WSP)

Crampton considered the case where in all constraints (ρ, S_1, S_2), S_1 and S_2 are singletons (and there were some extra restrictions).

Wang and Li considered, WSP(ρ^*), which is the case when all constraints (ρ, S_1, S_2) have the property that S_1 is a singleton and $\rho \in \rho^*$.

Workflow Satisfiability Problem (WSP)

Crampton considered the case where in all constraints (ρ, S_1, S_2), S_1 and S_2 are singletons (and there were some extra restrictions).

Wang and Li considered, WSP(ρ^*), which is the case when all constraints (ρ, S_1, S_2) have the property that S_1 is a singleton and $\rho \in \rho^*$.

- WSP($=$) is polynomial-time solvable

Workflow Satisfiability Problem (WSP)

Crampton considered the case where in all constraints (ρ, S_1, S_2), S_1 and S_2 are singletons (and there were some extra restrictions).

Wang and Li considered, WSP(ρ^*), which is the case when all constraints (ρ, S_1, S_2) have the property that S_1 is a singleton and $\rho \in \rho^*$.

- WSP($=$) is polynomial-time solvable
- WSP($\neq$) is NP-complete even if only type $(\neq, \{s'\}, \{s''\})$ constraints are used.

Known Results and Observations

As the problem in NP-hard (even in very simple cases!), Wang and Li considered it from the point og view of FPT.

Known Results and Observations

As the problem in NP-hard (even in very simple cases!), Wang and Li considered it from the point og view of FPT.

$k = |S|$ is small compared with $n = |U|$ and so k-WSP is of interest.

Wang and Li proved the following,

Known Results and Observations

As the problem in NP-hard (even in very simple cases!), Wang and Li considered it from the point og view of FPT.

$k = |S|$ is small compared with $n = |U|$ and so k-WSP is of interest.

Wang and Li proved the following,

- Complexity of k-WSP$(=, \neq)$ is $O^*(k^{k+1}(k-1)^{k2^{k-1}}) = O^*(2^{2^k k \log k})$

Known Results and Observations

As the problem in NP-hard (even in very simple cases!), Wang and Li considered it from the point og view of FPT.

$k = |S|$ is small compared with $n = |U|$ and so k-WSP is of interest.

Wang and Li proved the following,

- Complexity of k-WSP$(=, \neq)$ is $O^*(k^{k+1}(k-1)^{k2^{k-1}}) = O^*(2^{2^k k \log k})$
- Complexity of k-WSP$(\neq)$ is $O^*(k^{k+1}) = O^*(2^{k \log k})$

Results for Similar Problem

Fellows, Friedrich, Hermelin, Narodytska and Rosamond (IJCAI 2011):

- k-WSP($\neq$) with all constraints having only singletons

Results for Similar Problem

Fellows, Friedrich, Hermelin, Narodytska and Rosamond (IJCAI 2011):

- k-WSP($\neq$) with all constraints having only singletons
- Complexity is $O^*(kk!) = O^*(2^{k \log k})$

Outline

New Results: FPT

- Complexity of k-WSP$(=,\neq)$ is $O^*(2^k)$ (it was $O^*(2^{k\log k})$ for k-WSP$(\neq)$).

New Results: FPT

- Complexity of k-WSP$(=,\neq)$ is $O^*(2^k)$ (it was $O^*(2^{k\log k})$ for k-WSP$(\neq)$).
- This can be extended to some relations ρ added to $=,\neq$

New Results: FPT

- Complexity of k-WSP$(=,\neq)$ is $O^*(2^k)$ (it was $O^*(2^{k\log k})$ for k-WSP$(\neq)$).
- This can be extended to some relations ρ added to $=,\neq$
- Complexity cannot be decreased to $O^*(2^{o(k)})$ unless ETH fails

New Results: Kernels

- If all constraints use only singletons then there exist a kernel with at most k users

New Results: Kernels

- If all constraints use only singletons then there exist a kernel with at most k users

- k-WSP$(=, \neq)$ has no poly-size kernel unless $NP \subseteq coNP/poly$

New Results: Extensions

- If we do not restrict S_1, S_2 in constraints (ρ, S_1, S_2) then the problem is still FPT.

New Results: Extensions

- If we do not restrict S_1, S_2 in constraints (ρ, S_1, S_2) then the problem is still FPT.
- We can add a natural hierarchy and still remain FPT (which will be discussed later).

New Results: Extensions

- If we do not restrict S_1, S_2 in constraints (ρ, S_1, S_2) then the problem is still FPT.
- We can add a natural hierarchy and still remain FPT (which will be discussed later).
- With our hierarchy there will not exists polynomial kernels though (even if all constraints are of the form $(\neq, \{s_1\}, \{s_2\})$, unless $NP \subseteq coNP/poly$.

Outline

Max Weighted Partition Theorem

Björklund, Husfeldt and Koivisto (SIAM J. Comput., 2009):

Max Weighted Partition Theorem

Björklund, Husfeldt and Koivisto (SIAM J. Comput., 2009):

An *n-partition* of S is an n-tuple $(F_1, \ldots, F_n)$ s.t. $F_1 \cup \cdots \cup F_n = S$ and $F_i \cap F_j = \emptyset$ for all $i \neq j \in [n]$. Some blocks F_i can be empty.

Max Weighted Partition Theorem

Björklund, Husfeldt and Koivisto (SIAM J. Comput., 2009):

An *n-partition* of S is an n-tuple $(F_1, \ldots, F_n)$ s.t. $F_1 \cup \cdots \cup F_n = S$ and $F_i \cap F_j = \emptyset$ for all $i \neq j \in [n]$. Some blocks F_i can be empty.

MAX WEIGHTED PARTITION

Input: A set S of k elements and n functions ϕ_i, $i \in [n]$, from 2^S to integers from the range $[-M, M]$ $(M \geq 1)$.

Output: An n-partition $(F_1, \ldots, F_n)$ of S that maximizes $\sum_{i=1}^{n} \phi_i(F_i)$.

Max Weighted Partition Theorem

Björklund, Husfeldt and Koivisto (SIAM J. Comput., 2009):

An *n-partition* of S is an n-tuple $(F_1, \ldots, F_n)$ s.t. $F_1 \cup \cdots \cup F_n = S$ and $F_i \cap F_j = \emptyset$ for all $i \neq j \in [n]$. Some blocks F_i can be empty.

MAX WEIGHTED PARTITION
Input: A set S of k elements and n functions ϕ_i, $i \in [n]$, from 2^S to integers from the range $[-M, M]$ $(M \geq 1)$.
Output: An n-partition $(F_1, \ldots, F_n)$ of S that maximizes $\sum_{i=1}^{n} \phi_i(F_i)$.

Theorem

MAX WEIGHTED PARTITION *can be solved in time* $\widetilde{O}(2^k n^2 M)$.

$O^*(2^k)$ for k-WSP$(=,\neq)$: Ideas

- Partition S into blocks, each of which is allocated to a single (authorized) user.

$O^*(2^k)$ for k-WSP($=, \neq$): Ideas

- Partition S into blocks, each of which is allocated to a single (authorized) user.
- Find a partition in which each user is authorized to perform all steps in the block to which he/she is assigned.

$O^*(2^k)$ for k-WSP$(=,\neq)$: Ideas

- Partition S into blocks, each of which is allocated to a single (authorized) user.
- Find a partition in which each user is authorized to perform all steps in the block to which he/she is assigned.

We have constraints $(\neq, S_1, S_2)$ and $(=, S_1', S_2')$, where S_1 and S_1' are singletons. Then $F \subseteq S$ cannot be a block if

$O^*(2^k)$ for k-WSP$(=, \neq)$: Ideas

- Partition S into blocks, each of which is allocated to a single (authorized) user.
- Find a partition in which each user is authorized to perform all steps in the block to which he/she is assigned.

We have constraints $(\neq, S_1, S_2)$ and $(=, S_1', S_2')$, where S_1 and S_1' are singletons. Then $F \subseteq S$ cannot be a block if

- $F \supseteq S_1 \cup S_2$ or

$O^*(2^k)$ for k-WSP$(=,\neq)$: Ideas

- Partition S into blocks, each of which is allocated to a single (authorized) user.
- Find a partition in which each user is authorized to perform all steps in the block to which he/she is assigned.

We have constraints $(\neq, S_1, S_2)$ and $(=, S'_1, S'_2)$, where S_1 and S'_1 are singletons. Then $F \subseteq S$ cannot be a block if

- $F \supseteq S_1 \cup S_2$ or
- $F \cap (S'_1 \cup S'_2) = S'_i$, for some $i \in [2]$.

$O^*(2^k)$ for k-WSP$(=,\neq)$: New Result

- For each $F \subseteq S$, $\phi_i(F) = 1$ if $F = \emptyset$ or F can be a block and $u_i \in A(s_j)$, for all $s_j \in F$.

$O^*(2^k)$ for k-WSP$(=,\neq)$: New Result

- For each $F \subseteq S$, $\phi_i(F) = 1$ if $F = \emptyset$ or F can be a block and $u_i \in A(s_j)$, for all $s_j \in F$.
- Otherwise, $\phi_i(F) = 0$

$O^*(2^k)$ for k-WSP$(=,\neq)$: New Result

- For each $F \subseteq S$, $\phi_i(F) = 1$ if $F = \emptyset$ or F can be a block and $u_i \in A(s_j)$, for all $s_j \in F$.
- Otherwise, $\phi_i(F) = 0$

$O^*(2^k)$ for k-WSP$(=,\neq)$: New Result

- For each $F \subseteq S$, $\phi_i(F) = 1$ if $F = \emptyset$ or F can be a block and $u_i \in A(s_j)$, for all $s_j \in F$.
- Otherwise, $\phi_i(F) = 0$
- Using the Max Weighted Partition Theorem decide: is there a partition of weight n?

Lower Bound for k-WSP$(=, \neq)$: NAE-3-Sat

Not-All-Equal-3-Sat (NAE-3-Sat)

Input: A 3-CNF formula ϕ.

Output: Decide whether there is a truth assignment s.t. in every clause of ϕ at least one literal in TRUE and one is FALSE.

Lower Bound for k-WSP$(=, \neq)$: NAE-3-Sat

Not-All-Equal-3-Sat (NAE-3-Sat)
Input: A 3-CNF formula ϕ.
Output: Decide whether there is a truth assignment s.t. in every clause of ϕ at least one literal in TRUE and one is FALSE.

Lemma

Assuming ETH, there is $\epsilon > 0$ s.t. $\mathrm{NAE\text{-}3\text{-}SAT}$ with n variables cannot be solved in time $O(2^{\epsilon n})$.

Lower Bound for k-WSP$(=, \neq)$: NAE-3-Sat

NOT-ALL-EQUAL-3-SAT (NAE-3-SAT)

Input: A 3-CNF formula ϕ.

Output: Decide whether there is a truth assignment s.t. in every clause of ϕ at least one literal in TRUE and one is FALSE.

Lemma

Assuming ETH, there is $\epsilon > 0$ s.t. NAE-3-SAT with n variables cannot be solved in time $O(2^{\epsilon n})$.

Theorem

Even if $n = 2$, k-WSP$(\neq)$ cannot be solved in time $O^*(2^{\epsilon k})$ for some $\epsilon > 0$ unless ETH fails.

Lower Bound for k-WSP$(=, \neq)$: Using NAE-3-Sat

Theorem

Even if $n = 2$, k-WSP$(\neq)$ cannot be solved in time $O^*(2^{\epsilon k})$ for some real $\epsilon > 0$ unless ETH fails.

Lower Bound for k-WSP$(=, \neq)$: Using NAE-3-Sat

Theorem

Even if $n = 2$, k-WSP$(\neq)$ cannot be solved in time $O^*(2^{\epsilon k})$ for some real $\epsilon > 0$ unless ETH fails.

- instance of NAE-3-Sat with variables $x_1, \ldots, x_t$.

Lower Bound for k-WSP$(=, \neq)$: Using NAE-3-Sat

Theorem

Even if $n = 2$, k-WSP($\neq$) cannot be solved in time $O^*(2^{\epsilon k})$ for some real $\epsilon > 0$ unless ETH fails.

- instance of NAE-3-Sat with variables $x_1, \ldots, x_t$.
- set $s_i := x_i$, $s_{t+i} := \bar{x}_i$ and $k := 2t$
- add constraint $(\neq, s_i, s_{t+i})$

Lower Bound for k-WSP$(=, \neq)$: Using NAE-3-Sat

Theorem

Even if $n = 2$, k-WSP$(\neq)$ cannot be solved in time $O^*(2^{\epsilon k})$ for some real $\epsilon > 0$ unless ETH fails.

- instance of NAE-3-Sat with variables $x_1, \ldots, x_t$.
- set $s_i := x_i$, $s_{t+i} := \bar{x}_i$ and $k := 2t$
- add constraint $(\neq, s_i, s_{t+i})$
- for a clause with literals s_i, s_j, s_q we add constraint $(\neq, s_i, \{s_j, s_q\})$

Lower Bound for k-WSP($=, \neq$): Using NAE-3-Sat

Theorem

Even if $n = 2$, k-WSP($\neq$) cannot be solved in time $O^*(2^{\epsilon k})$ for some real $\epsilon > 0$ unless ETH fails.

- instance of NAE-3-Sat with variables $x_1, \ldots, x_t$.
- set $s_i := x_i$, $s_{t+i} := \bar{x}_i$ and $k := 2t$
- add constraint $(\neq, s_i, s_{t+i})$
- for a clause with literals s_i, s_j, s_q we add constraint $(\neq, s_i, \{s_j, s_q\})$
- USER1=TRUE, USER2=FALSE

Outline

Organizational Hierarchies: Definitions

- Let $(X_1, \ldots, X_p)$ and $(Y_1, \ldots Y_q)$ be p- and q-partitions of the same set.

Organizational Hierarchies: Definitions

- Let $(X_1, \ldots, X_p)$ and $(Y_1, \ldots Y_q)$ be p- and q-partitions of the same set. We say that $(Y_1, \ldots Y_q)$ is a *refinement* of $(X_1, \ldots, X_p)$ if for each $i \in [q]$ there exists $j \in [p]$ s.t. $Y_i \subseteq X_j$.

Organizational Hierarchies: Definitions

- Let $(X_1, \ldots, X_p)$ and $(Y_1, \ldots Y_q)$ be p- and q-partitions of the same set. We say that $(Y_1, \ldots Y_q)$ is a *refinement* of $(X_1, \ldots, X_p)$ if for each $i \in [q]$ there exists $j \in [p]$ s.t. $Y_i \subseteq X_j$.
- Let U be the set of users in an organization. An *organizational ℓ-hierarchy* is a collection of ℓ partitions of U, $\mathcal{H} = U^{(1)}, \ldots, U^{(\ell)}$, where $U^{(i)}$ is a refinement of $U^{(i+1)}$.

Organizational Hierarchies: Definitions

- Let $(X_1, \ldots, X_p)$ and $(Y_1, \ldots Y_q)$ be p- and q-partitions of the same set. We say that $(Y_1, \ldots Y_q)$ is a *refinement* of $(X_1, \ldots, X_p)$ if for each $i \in [q]$ there exists $j \in [p]$ s.t. $Y_i \subseteq X_j$.
- Let U be the set of users in an organization. An *organizational ℓ-hierarchy* is a collection of ℓ partitions of U, $\mathcal{H} = U^{(1)}, \ldots, U^{(\ell)}$, where $U^{(i)}$ is a refinement of $U^{(i+1)}$.
- We say $\mathcal{H}$ is *canonical* if it satisfies the following:
  - $U^{(i)} \neq U^{(i+1)}$;
  - $U^{(\ell)}$ is a 1-partition containing the set U;
  - $U^{(1)}$ is an n-partition containing every singleton from U.

Organizational Hierarchies: Theorem

- $u_p \sim_i u_q$ iff $u_p, u_q \in$ same block of $U^{(i)}$

Organizational Hierarchies: Theorem

- $u_p \sim_i u_q$ iff $u_p, u_q \in$ same block of $U^{(i)}$
- $u_p \not\sim_i u_q$ iff $u_p, u_q \in$ different blocks of $U^{(i)}$

Organizational Hierarchies: Theorem

- $u_p \sim_i u_q$ iff $u_p, u_q \in$ same block of $U^{(i)}$
- $u_p \not\sim_i u_q$ iff $u_p, u_q \in$ different blocks of $U^{(i)}$

Theorem

Given a canonical organizational ℓ-hierarchy, k-WSP$(\sim_1, \not\sim_1, \ldots, \sim_\ell, \not\sim_\ell)$ can be solved in time $O^*(3^k)$.

Organizational Hierarchies: Theorem

- $u_p \sim_i u_q$ iff $u_p, u_q \in$ same block of $U^{(i)}$
- $u_p \not\sim_i u_q$ iff $u_p, u_q \in$ different blocks of $U^{(i)}$

Theorem

Given a canonical organizational ℓ-hierarchy, k-WSP$(\sim_1, \not\sim_1, \ldots, \sim_\ell, \not\sim_\ell)$ can be solved in time $O^*(3^k)$.

Theorem

Given a canonical organizational 3-hierarchy, k-WSP$(\sim_1, \not\sim_1, \ldots, \sim_3, \not\sim_3)$ has no polynomial kernel unless NP$\subseteq$ coNP/poly (even if all constraints only contain singletons).

The End

- Thank you!
- Questions?